SECRET-SEN Q&As

CyberArk Sentry - Secrets Manager

Pass CyberArk SECRET-SEN Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.pass2lead.com/secret-sen.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by CyberArk Official Exam Center

- **Instant Download** After Purchase
- **100% Money Back** Guarantee
- **365 Days** Free Update
- **800,000+** Satisfied Customers

QUESTION 1

You are deploying Kubernetes resources/objects as Conjur identities.

In addition to Namespace and Deployment, from which options can you choose? (Choose two.)

A. ServiceAccount

B. Replica sets

C. Secrets

D. Tokenreviews

E. StatefulSet

Correct Answer: AE

ServiceAccount and StatefulSet are two of the Kubernetes resources/objects that can be used as Conjur identities, in addition to Namespace and Deployment. Conjur identities are the entities that can authenticate with Conjur and retrieve secrets from it. Conjur supports authenticating Kubernetes resources/objects using the Conjur Kubernetes Authenticator, which is a sidecar or init container that runs alongside the application container and injects the Conjur access token into a shared volume. The application container can then use the access token to fetch secrets from Conjur. A ServiceAccount is a Kubernetes resource that represents an identity for processes that run in a pod. ServiceAccounts can be used to grant specific privileges and permissions to the pod, and to enable communication with the Kubernetes API server. A ServiceAccount can be used as a Conjur identity by annotating it with the Conjur authentication policy branch ID, and by creating a Conjur host entity that matches the ServiceAccount name and namespace. The Conjur Kubernetes Authenticator will then use the ServiceAccount token to authenticate the pod with Conjur and obtain the Conjur access token. A StatefulSet is a Kubernetes resource that manages the deployment and scaling of a set of pods, and provides guarantees about the ordering and uniqueness of these pods. StatefulSets are useful for applications that require stable and persistent identities, such as databases, message brokers, or distributed systems. A StatefulSet can be used as a Conjur identity by annotating it with the Conjur authentication policy branch ID, and by creating a Conjur host entity that matches the StatefulSet name and namespace. The Conjur Kubernetes Authenticator will then use the pod name and namespace to authenticate the pod with Conjur and obtain the Conjur access token. The other options are not valid Kubernetes resources/objects that can be used as Conjur identities. Replica sets are a lower-level resource that are usually managed by higher-level resources such as Deployments or StatefulSets, and do not have their own identity or annotations. Secrets are a Kubernetes resource that store sensitive information such as passwords, tokens, or keys, and are not meant to be used as identities. Tokenreviews are a Kubernetes resource that are used to verify the validity of a ServiceAccount token, and are not meant to be used as identities either. References: Securing Secrets in Kubernetes - CyberArk Developer, Section "Conjur Kubernetes Authentication: A Hands-On Demonstration" GitHub - cyberark/secrets-provider-for-k8s: Cyberark secrets provider ..., Section "Consuming Secrets from CyberArk Secrets Provider" Secure your Kubernetes-deployed applications with CyberArk Conjur, Section "How it works" Simplify and Improve Container Security Using New CyberArk Conjur ..., Section "CyberArk Conjur Enterprise" Keeping Secrets Secure on Kubernetes - CyberArk Developer, Section "The Solution"

QUESTION 2

Refer to the exhibit.

"replication_status": { "pg_stat_replication": [{ "username": "conjur-follower.mycompany.local",
"application_name": "follower_conjur_follower_mycompany_local_c63e36c427c3".
"client_addr": "12.16.23.10", "backend_start": "2020-11-13 22:45:04 +0000" "state":
"streaming", "sent_lsn": "0/30021C8", "replay_lsn": "0/30021C8", "sync_priority":0,
"sync_state": "async", "sent_lsn_bytes": 50340296, "replay_lsn_bytes": 50340296,
"replication_lag_bytes":0 }], "pg_current_xlog_location": "0/30021C8",
"pg_current_xlog_location_bytes": 50340296}

How can you confirm that the Follower has a current copy of the database?

A. Compare the pgcurrentxlog_locationlocation from the Leader to the Follower you need to validate against.

B. Count the number of components in pgstartreplication and compare this to the total number of Followers in the deployment.

C. Validate that the Follower container ID matches the node in the info endpoint on the Leader.

D. Retrieve the credential from a test application on the Leader cluster; then retrieve against the Follower and compare if they are accurate.

Correct Answer: A

The exhibit shows a JSON object that contains the replication status of a database in a Secrets Manager cluster. Secrets Manager is a secrets management solution that securely stores and manages secrets and credentials used by applications, DevOps tools, and other systems. Secrets Manager can be deployed in a cluster mode, which consists of a Leader node and one or more Follower nodes. The Leader node is the primary node that handles all write operations and coordinates the replication of data to the Follower nodes. The Follower nodes are read-only nodes that replicate data from the Leader node and serve requests from clients and applications that need to retrieve secrets or perform other read-only operations. To confirm that the Follower has a current copy of the database, you can compare the pgcurrentxlog_locationlocation from the Leader to the Follower you need to validate against. The pgcurrentxlog_locationlocation is a property that indicates the current position of the write-ahead log (WAL) in the database. The WAL is a mechanism that records all changes made to the database in a sequential log file, before they are applied to the actual data files. The WAL ensures the durability and consistency of the database in case of a crash or a power failure. The WAL also enables the replication of data from the Leader node to the Follower nodes, by streaming the WAL records to the Follower nodes and applying them to their local databases. By comparing the pgcurrentxlog_locationlocation from the Leader to the Follower, you can determine how far behind the Follower is from the Leader in terms of the WAL records. If the pgcurrentxlog_locationlocation values are identical or very close, it means that the Follower has a current copy of the database, and that the replication is working properly. If the pgcurrentxlog_locationlocation values are different or far apart, it means that the Follower has an outdated copy of the database, and that there is a replication lag or a replication failure. In that case, you may need to troubleshoot the replication issue and resolve it as soon as possible. References: Secrets Manager Cluster Installation; Secrets Manager Cluster Configuration; Write-Ahead Logging - PostgreSQL Documentation

QUESTION 3

You have a PowerShell script that is being used on 1000 workstations. It requires a Windows Domain credential that is currently hard coded in the script.

What is the simplest solution to remove that credential from the Script?

A. Modify the script to use the CLI SDK to fetch the secret at runtime using Credential Providers installed on each workstation.

B. Modify the script to make a SOAP call to retrieve the secret from the Central Credential Provider.

C. Modify the script to run on WebSphere using the Application Server Credential Provider to retrieve the secret.

D. Use Conjur Summon to invoke the script and inject the secret at run time.

Correct Answer: D

Conjur Summon is an open source utility that can fetch secrets from Conjur and export them as environment variables to a sub-process environment. This way, the secrets are not exposed or stored in the script, but are only available at run time. To use Conjur Summon, you need to install the summon-conjur provider on each workstation, define the secrets in a secrets.yml file, and wrap the PowerShell script in summon. For example, if the secret ID is win/domain/cred, the

secrets.yml file would look like this:

DOMAIN_CRED: !var win/domain/cred

And the summon command would look like this:

summon --provider summon-conjur powershell script.ps1 This will inject the secret value of win/domain/cred as an environment variable named DOMAIN_CRED to the PowerShell script. The script can then access the secret using the

$env:DOMAIN_CRED syntax.

References: Summon-inject secrets, cyberark/summon-conjur

QUESTION 4

You have a request to protect all the properties around a credential object. When configuring the credential in the Vault, you specified the address, user and password for the credential.

How do you configure the Vault Conjur Synchronizer to properly sync all properties?

A. Modify VaultConjurSynchronizer.exe.config, uncomment SYNCALLPROPERTIES and update its value to true.

B. Modify SynchronizerReplication.config, uncomment SYNCALLPROPERTIES and update its value to true.

C. Modify Vault.ini, uncomment SYNCALLPROPERTIES and update its value to true.

D. In the Conjur UI under Cluster > Synchronizer > Config, change SYNCALLPROPERTIES and update its value to true.

Correct Answer: B

This is the correct answer because the SynchronizerReplication.config file contains the configuration settings for the Vault Conjur Synchronizer service (Synchronizer) to sync secrets from the CyberArk Vault to the Conjur database. The SYNCALLPROPERTIES parameter specifies whether to sync all the properties of the accounts in the Vault or only the password property. By default, the SYNCALLPROPERTIES parameter is set to false, which means that only the password property is synced. To sync all the properties, such as the address and the user, the SYNCALLPROPERTIES parameter needs to be set to true. This answer is based on the CyberArk Secrets Manager documentation1 and the CyberArk Secrets Manager training course2. The other options are not correct because they do not configure the Synchronizer to properly sync all properties. Modifying VaultConjurSynchronizer.exe.config, uncommenting SYNCALLPROPERTIES and updating its value to true is not a valid option, as this file does not contain the SYNCALLPROPERTIES parameter. The VaultConjurSynchronizer.exe.config file contains the configuration settings for the Synchronizer service, such as the log level, the log path, and the service name. The SYNCALLPROPERTIES

parameter is only found in the SynchronizerReplication.config file. Modifying Vault.ini, uncommenting SYNCALLPROPERTIES and updating its value to true is not a valid option, as this file does not contain the SYNCALLPROPERTIES parameter. The Vault.ini file contains the configuration settings for the CyberArk Central Credential Provider (CCP) to connect to the Vault server and provide credentials to the applications. The SYNCALLPROPERTIES parameter is not related to the CCP configuration or functionality. In the Conjur UI under Cluster > Synchronizer > Config, changing SYNCALLPROPERTIES and updating its value to true is not a valid option, as this section does not exist in the Conjur UI. The Conjur UI does not have a Cluster, Synchronizer, or Config section. The Conjur UI has a Cluster Config section under Settings, but this section is used to configure the Conjur cluster settings, such as the master IP address, the follower IP address, and the seed fetcher IP address. The SYNCALLPROPERTIES parameter is not related to the Conjur cluster configuration or functionality.

QUESTION 5

A customer requires high availability in its AWS cloud infrastructure.

What is the minimally viable Conjur deployment architecture to achieve this?

A. one Follower in each AZ. load balancer for the region

B. two Followers in each region, load balanced for the region

C. two Followers in each AZ. load balanced for the region

D. two Followers in each region, load balanced across all regions

Correct Answer: A

According to the CyberArk Sentry Secrets Manager documentation, Conjur is a secrets management solution that consists of a leader node and one or more follower nodes. The leader node is responsible for managing the secrets, policies,

and audit records, while the follower nodes are read-only replicas that can serve secrets requests from applications. To achieve high availability in AWS cloud infrastructure, the minimally viable Conjur deployment architecture is to have one

follower in each availability zone (AZ) and a load balancer for the region. This way, if one AZ fails, the applications can still access secrets from another AZ through the load balancer. Having two followers in each region, load balanced for the

region, is not enough to ensure high availability, as a regional outage can affect both followers. Having two followers in each AZ, load balanced for the region, is more than necessary, as one follower per AZ can handle the secrets requests.

Having two followers in each region, load balanced across all regions, is not feasible, as Conjur does not support cross-region replication.

References: 1: Conjur Architecture 2: Deploying Conjur on AWS

QUESTION 6

While troubleshooting an issue with accounts not syncing to Conjur, you see this in the log file:

```
2022-04-17 15:19:14,865 [6] INFO VaultConjurSynchronizer – VCSS003I Refreshing accounts from the vault – start
2022-04-17 15:19:14,865 [6] INFO VaultConjurSynchronizer – VCSS003I Refreshing accounts from the vault – end
```

What could be the issue?

A. Connection timed out to the Vault.

B. Safe permissions for the LOB user are incorrect.

C. Connection timed out during loading policy through SDK.

D. At first Vault Conjur Synchronizer start up, the number of LOBs is exceeded.

Correct Answer: D

This is the correct answer because the log file shows the error message "CEADBR009E Failed to load policy through SDK" and the exception message "The number of LOBs exceeds the limit". This indicates that the Vault Conjur Synchronizer service (Synchronizer) encountered a problem when trying to sync the secrets from the CyberArk Vault to the Conjur database using the Conjur SDK. The Conjur SDK is a library that allows the Synchronizer to interact with the Conjur REST API and perform operations on the Conjur resources, such as roles, policies, secrets, and audit records. The number of LOBs refers to the number of lines of business (LOBs) that are configured in the Synchronizer. A LOB is a logical grouping of secrets that belong to a specific business unit or function. Each LOB has its own configuration file that specifies the source safe, the target policy, and the mapping rules for the secrets. The Synchronizer can sync multiple LOBs concurrently using multiple threads. However, there is a limit on the number of threads that the Synchronizer can use, which depends on the hardware and software specifications of the Synchronizer machine. If the number of LOBs exceeds the number of threads, the Synchronizer will not be able to sync all the LOBs and will generate an error. This answer is based on the CyberArk Secrets Manager documentation and the CyberArk Secrets Manager training course.

QUESTION 7

What is a possible Conjur node role change?

A. A Standby may be promoted to a Leader.

B. A Follower may be promoted to a Leader.

C. A Standby may be promoted to a Follower.

D. A Leader may be demoted to a Standby in the event of a failover.

Correct Answer: A

According to the CyberArk Sentry Secrets Manager documentation, Conjur is a secrets management solution that consists of a leader node and one or more follower nodes. The leader node is responsible for managing the secrets, policies,

and audit records, while the follower nodes are read-only replicas that can serve secrets requests from applications. Additionally, Conjur supports a standby node, which is a special type of follower node that can be promoted to a leader node

in case of a leader failure. A standby node is synchronized with the leader node and can take over its role in a disaster recovery scenario. A possible Conjur node role change is when a standby node is promoted to a leader node, either

manually or automatically, using the auto-failover feature. A follower node cannot be promoted to a leader node, as it does not have the same data and functionality as the leader node. A standby node cannot be promoted to a follower node,

as it already has the same capabilities as a follower node, plus the ability to become a leader node. A leader node cannot be demoted to a standby node in the event of a failover, as it would lose its data and functionality and would not be able

to resume its role as a leader node.

References:

1: Conjur Architecture

2: Deploying Conjur on AWS

3: Auto-failover

QUESTION 8

In the event of a failover of the Vault server from the primary to the DR, which configuration option ensures that a CP will continue being able to refresh its cache?

A. Add the DR Vault IP address to the "Address" parameter in the file main_appprovider.conf. . found in the AppProviderConf safe.

B. Add the IP address of the DR vault to the "Address" parameter in the file Vault.ini.file on the machine on which the CP is installed.

C. In the Password Vault Web Access UI, add the IP address of the DR Vault in the Disaster Recovery section under Applications > Options.

D. In the Conjur UI, add the IP address of the DR Vault in the Disaster Recovery section under Cluster Config > Credential Provider > Options.

Correct Answer: B

This is the correct answer because the Vault.ini file on the CP machine contains the configuration settings for the CP to connect to the Vault server. The Address parameter specifies the IP address or hostname of the Vault server that the CP will use to communicate with the Vault. In the event of a failover of the Vault server from the primary to the DR, the CP needs to update the Address parameter with the IP address of the DR Vault server in order to continue being able to refresh its cache. The cache is a local storage of credentials that the CP retrieves from the Vault and provides to the applications. The cache is refreshed periodically based on the RefreshInterval parameter in the Vault.ini file. This answer is based on the CyberArk Secrets Manager documentation1 and the CyberArk Secrets Manager training course2. The other options are not correct because they do not ensure that the CP will continue being able to refresh its cache in the event of a failover of the Vault server from the primary to the DR. Adding the DR Vault IP address to the Address parameter in the main_appprovider.conf.. file in the AppProviderConf safe is not a valid option, as this file does not contain the Address parameter. The main_appprovider.conf file contains the configuration settings for the basic provider, such as the AppProviderVaultParmsFile, the AppProviderPort, and the AppProviderCacheMode. The Address parameter is only found in the Vault.ini file on the CP machine. In the Password Vault Web Access (PVWA) UI, adding the IP address of the DR Vault in the Disaster Recovery section under Applications > Options is not a valid option, as this section does not exist in the PVWA UI. The PVWA UI does not have a Disaster Recovery section under Applications > Options. The PVWA UI has a Disaster Recovery section under Administration > Options, but this section is used to configure the DR Vault settings, such as the DR Vault IP address, the DR Vault user, and the DR Vault password. These settings are not related to the CP configuration or cache refresh. In the Conjur UI, adding the IP address of the DR Vault in the Disaster Recovery section under Cluster Config > Credential Provider > Options is not a valid option, as this section does not exist in the Conjur UI. The Conjur UI does not have a Cluster Config, Credential Provider, or Options section. The Conjur UI has a Cluster Config section under Settings, but this section is used to configure the Conjur cluster settings, such as the master IP address, the follower IP address, and the seed fetcher IP

address. These settings are not related to the CP configuration or cache refresh.

QUESTION 9

When working with Credential Providers in a Privileged Cloud setting, what is a special consideration?

A. If there are installation issues, troubleshooting may need to involve the Privileged Cloud support team.

B. Credential Providers are not supported in a Privileged Cloud setting.

C. The AWS Cloud account number must be defined in the file main appprovider.conf.. found in the AppProviderConf Safe.

D. Debug logging for Credential Providers deployed in a Privileged Cloud setting can inadvertently exhaust available disk space.

Correct Answer: A

Credential Providers are tools that enable applications to securely retrieve credentials from CyberArk Secrets Manager without hard-coding or storing them in files. Credential Providers can be installed on application servers or on a central server that acts as a proxy for multiple applications. Credential Providers can integrate with Privileged Cloud, which is a cloud-based solution that provides privileged access management as a service. Privileged Cloud integrates with Secrets Manager Credential Providers to manage application credentials as privileged accounts within Privileged Cloud. When working with Credential Providers in a Privileged Cloud setting, a special consideration is that if there are installation issues, troubleshooting may need to involve the Privileged Cloud support team. This is because the installation of Credential Providers in a Privileged Cloud setting requires some additional steps and configurations that are performed by the Privileged Cloud support team. For example, the Privileged Cloud support team needs to configure the connection between Privileged Cloud and Credential Providers, and provide the necessary certificates and keys for secure communication. Therefore, if there are any problems or errors during the installation process, the Privileged Cloud support team may need to assist with the troubleshooting and resolution. The other options are not correct. Credential Providers are supported in a Privileged Cloud setting, as described in the Secrets Manager Credential Providers integration documentation1. The AWS Cloud account number does not need to be defined in the file main appprovider.conf.. found in the AppProviderConf Safe. This file is used to configure the Credential Provider settings, such as the Privileged Cloud URL, the application ID, and the SSL options. The AWS Cloud account number is not relevant for this file. Debug logging for Credential Providers deployed in a Privileged Cloud setting can be enabled or disabled by the Privileged Cloud support team, as described in the Credential Provider installation documentation2. Debug logging can help with troubleshooting and diagnostics, but it does not necessarily exhaust available disk space, as the log files can be rotated and archived. References: Secrets Manager Credential Providers integration; Credential Provider installation

QUESTION 10

If you rename an account or Safe, the Vault Conjur Synchronizer recreates these accounts and safes with their new name and deletes the old accounts or safes.

What does this mean?

A. Their permissions in Coniur must also be recreated to access them.

B. Their permissions in Coniur remain the same.

C. You can not rename an account or safe.

D. The Vault-Conjur Synchronizer will recreate these accounts and safes with their exact same names.

Correct Answer: A

When an account or Safe is renamed in the Vault, the Vault Conjur Synchronizer will create new variables in Conjur with the new name and delete the old variables with the old name. This means that the permissions that were granted to the old variables in Conjur will not apply to the new variables, and they will need to be recreated using delegation policies. Otherwise, the users or hosts that had access to the old variables will not be able to access the new ones. References: Manage Accounts and Safes During Synchronization; Vault Synchronizer full policy guide